

# Estruturas de Dados - ESP412

Prof<sup>a</sup> Ana Carolina Sokolonski

Bacharelado em Sistemas de Informação  
Instituto Federal de Ciência e Tecnologia da Bahia  
Campus de Feira de Santana

*carolsoko@ifba.edu.br*

April 8, 2025

# Árvores B (B-Trees)

## 1 Árvores B (B-Trees)

- Definição
- Inserção em Árvores B
  - Tipo do Nó

## 2 Árvores B + (B+-Trees)

- Tipo do Nó

## 3 Aplicações - Árvores B e B+

## 4 Referências

# Árvores B (B-Trees)

# Árvores B (B-Trees)

## Definição:

Árvores B (B-Trees) são Árvores de Busca/Pesquisa Balanceadas, projetadas para funcionar bem em HD's e outros dispositivos de armazenamento secundário, por isso são usadas na maioria dos Sistemas de Arquivos e Bancos de Dados. As ÁRVORES B minimizam os acessos de E/S em disco.

# Árvore B (B-Trees)

## Definição:

Árvore B (B-Trees) são Árvores de Busca/Pesquisa Balanceadas, projetadas para funcionar bem em HD's e outros dispositivos de armazenamento secundário, por isso são usadas na maioria dos Sistemas de Arquivos e Bancos de Dados. As ÁRVORES B minimizam os acessos de E/S em disco.

As ÁRVORES B foram criadas por Bayes e McCreight em 1972. Este tipo de Árvore admite que seus nós tenham muitos filhos. A quantidade de filhos costuma variar de uma dezena até milhares. Esta quantidade chama-se “Fator de Ramificação”.  
[Cormen et al. 2009]

# Árvore B (B-Trees)

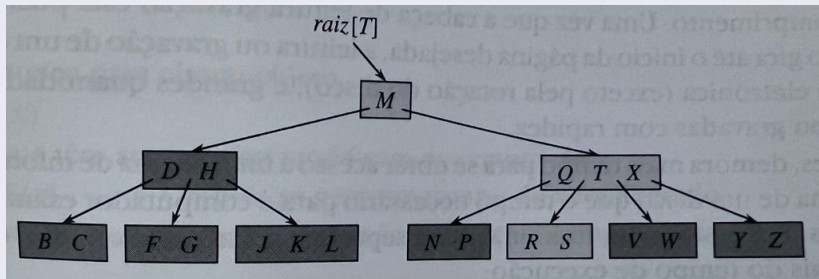
## Definição:

As ÁRVORES B generalizam as Árvores de Busca Binária de forma natural.

# Árvores B (B-Trees)

## Definição:

As ÁRVORES B generalizam as Árvores de Busca Binária de forma natural.



# Árvores B (B-Trees)

## Definição:

Uma **ÁRVORES B** obedece as propriedades a seguir:



# Árvores B (B-Trees)

## Definição:

Uma **ÁRVORES B** obedece as propriedades a seguir:

- 1 Todo nó  $x$  possui os seguintes campos:
  - $n[x]$ , o número de chaves atualmente armazenadas no nó  $x$ ;
  - As próprias  $n[x]$  chaves, armazenadas em ordem crescente;
  - $folha[x]$ , um valor booleano que é *TRUE* se  $x$  é folha e *FALSE* se  $x$  é galho ou raiz.

# Árvores B (B-Trees)

## Definição:

Uma **ÁRVORES B** obedece as propriedades a seguir:

- 1 Todo nó  $x$  possui os seguintes campos:
  - $n[x]$ , o número de chaves atualmente armazenadas no nó  $x$ ;
  - As próprias  $n[x]$  chaves, armazenadas em ordem crescente;
  - $folha[x]$ , um valor booleano que é *TRUE* se  $x$  é folha e *FALSE* se  $x$  é galho ou raiz.
- 2 Cada nó galho  $x$ , contém  $n[x] + 1$  ponteiros para seus filhos ( $c_1[x], c_2[x], \dots, c_{n[x]+1}[x]$ ). Os nós folha não têm filhos, por isso seus ponteiros  $c_i$  são nulos.

# Árvores B (B-Trees)

## Definição:

Uma ÁRVORES B obedece as propriedades a seguir:

- 3 As chaves  $chave_i[x]$  separam os intervalos de chaves armazenadas em cada subárvore:
  - Se  $k_i$  é qualquer chave armazenada na subárvore com raiz  $c_i[x]$ , então
$$k_1 \leq chave_1[x] \leq k_2 \leq chave_2[x] \leq .. \leq chave_{n[x]}[x] \leq k_{n[x]+1}$$

# Árvores B (B-Trees)

## Definição:

Uma ÁRVORES B obedece as propriedades a seguir:

- 3 As chaves  $chave_i[x]$  separam os intervalos de chaves armazenadas em cada subárvore:
  - Se  $k_i$  é qualquer chave armazenada na subárvore com raiz  $c_i[x]$ , então
$$k_1 \leq chave_1[x] \leq k_2 \leq chave_2[x] \leq .. \leq chave_{n[x]}[x] \leq k_{n[x]+1}$$
- 4 Toda folha tem a mesma profundidade, que é a altura  $h$  da árvore.

# Árvores B (B-Trees)

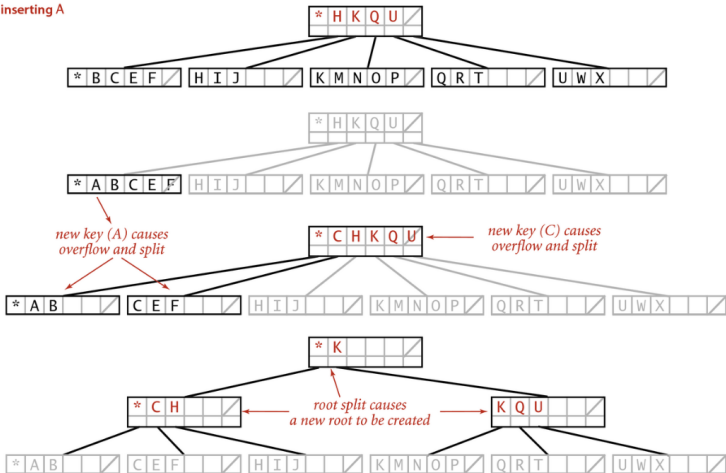
## Definição:

Uma ÁRVORES B obedece as propriedades a seguir:

- 5 Existem limites inferiores e superiores para a quantidade de chaves que um nó pode conter ( $t \geq 2$ ), que é o GRAU MÍNIMO da árvore.
  - Todo nó que não é raiz deve ter, NO MÍNIMO,  $t - 1$  chaves. Assim, todo galho tem, NO MÍNIMO,  $t$  filhos. Se a árvore não é vazia, a raiz tem ao menos uma chave.
  - Todo nó que não é raiz deve ter, NO MÁXIMO,  $2t - 1$  chaves. Assim, todo galho tem, NO MÁXIMO,  $2t$  filhos. Um nó é dito COMPLETO se ele tem exatamente  $2t - 1$  chaves.

# Inserção em Árvores B

inserting A



Inserting a new key into a B-tree set

# Árvore B

```
#define ORDEM 2

typedef struct chaveArvoreB{
    int chave;
    char nome[30];
    int folha;
}KEY;

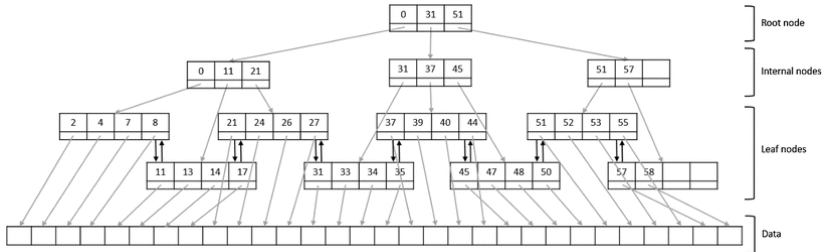
typedef struct paginaArvoreB *POINTER;

typedef struct paginaArvoreB{
    int numPag;
    KEY listaChaves[2*ORDEM];
    POINTER ponteiros[2*ORDEM + 1];
}PAG;
```

# Árvores B+ (B+-Trees)



# Árvores B+



# Aplicações - Árvores B e B+

## Aplicações

- o sistema de arquivos NTFS do Windows,
- o sistema de arquivos HFS e AHFS do Mac,
- os sistemas de arquivos JFS do Linux
- os bancos de dados ORACLE, DB2, INGRES, SQL e PostgreSQL
- entre outros...

# Referências

# Referências



CORMEN, T. H. et al. *Introduction to Algorithms*. 2nd. ed.  
[S.l.]: The MIT Press, 2009. ISBN 0262032937.